Attorney Docket No.:   3657-1013-1

PATENT

IN THE U.S. PATENT AND TRADEMARK OFFICE

Applicant: NILSEN et al.

Appl. No.:   10/615,928

Filed: July 10, 2003

For: SEAMLESS IP MOBILITY ACROSS SECURITY
                BOUNDARIES

L E T T E R

Assistant Commissioner for Patents   Date: August 12, 2003
P.O. Box 1450
Alexandria, Virginia 22313-1450

Sir:

Under the provisions of 35 U.S.C. § 119 and 37 C.F.R. § 1.55(a), the applicant(s) hereby claim(s) the right of priority based on the following application(s):

Country	Application No.	Filed
SWEDEN	20023336	July 11, 2002

A certified copy of the above-noted application(s) is(are) attached hereto.

If necessary, the Commissioner is hereby authorized in this, concurrent, and future replies, to charge payment or credit any overpayment to Deposit Account No. 25-0120 for any additional fee required under 37 C.F.R. §§ 1.16 or 1.17; particularly, extension of time fees.

Respectfully submitted,

YOUNG & THOMPSON

By /Benoît Castel/

Benoît Castel, #35,041

745 South 23rd Street, Suite 200
BC/psf   Arlington, Virginia 22202
                  (703) 521-2297

Attachment

(Rev. 04/19/2000)

KONGERIKET NORGE
The Kingdom of Norway

Bekreftelse på patentsøknad nr
Certification of patent application no

▽

2002 3336

▷ Det bekreftes herved at vedheftede dokument er nøyaktig utskrift/kopi av ovennevnte søknad, som opprinnelig inngitt 2002.07.11

▷ *It is hereby certified that the annexed document is a true copy of the above-mentioned application, as originally filed on 2002.07.11*

2003.07.04

Freddy Strømmen (signature)

Freddy Strømmen
Seksjonsleder

Line Reum (signature)

Line Reum

△ PATENTSTYRET ®
Styret for det industrielle rettsvern

PATENTSTYRET ®
Styret for det industrielle rettsvern

ADRESSE
► Postboks 8160 Dep.
Københavngaten 10
0033 Oslo

2002 -07- 11 0—

TELEFON
22 38 73 00
TELEFAKS
22 38 73

BANKGIRO
► 8276.01.00192
ORG. NR.

PATENTSTYRET

Søknad om patent

la - r

02-07-11*20023336

Skal utfylles av Patentstyret

Behandlende medlem **ER**
Int. Cl⁶ $H\ 04\ L$

Alm. tilgj. 12 JAN 2004

Søkers/fullmektigens referanse (angis hvis ønsket): O.nr. E26094 JFW/HBA	

Oppfinnelsens benevnelse: Sømløs Ip-mobilitet på tvers av sikkerhetsgrenser.

Hvis søknaden er en internasjonal søknad som videreføres etter patentlovens § 31:

Den internasjonale søknads nummer ....

Den internasjonale søknads inngivelsesdag ....

Søker:
Navn, bopel og adresse.
(Hvis patent søkes av flere: opplysning om hvem som skal være bemyndighet til å motta meddelelser fra Patentstyret på vegne av søkerne).

(Fortsett om nødvendig på neste side)

Birdstep Technology ASA
Haakon VII's gt. 5B
0161 Oslo

☐ Søker er en enkeltperson eller en småbedrift, eller flere slike i fellesskap med fast ansatte som til-sammen utfører 20 årsverk eller mindre (på søknadstidspunktet). Det er søkers ansvar å krysse av her for å oppnå laveste satser for søknadsavgift. NB! se også utfyllende forklaring på siste side.

Oppfinner:
Navn og (privat-) adresse
(Fortsett om nødvendig på neste side)

Frode Beckmann Nilsen, Konvallveien 51, 0875 Oslo
Espen Klovning, Gamle Strømsvei 23, 2010 Strømmen
Haakon Bryhni, Gørbitzgate 6, 0360 Oslo

Fullmektig:

ZACCO
BRYNS ZACCO AS
Karl J. ....ansgt. 25
P.O. Boks 765 Sentrum, N-0106 OSLO

Hvis søknad tidligere er inngitt i eller utenfor riket:
(Fortsett om nødvendig på neste side)

Prioritet kreves fra dato .... sted .... nr. ....

Prioritet kreves fra dato .... sted .... nr. ....

Prioritet kreves fra dato .... sted .... nr. ....

Hvis avdelt søknad: Den opprinnelige søknads nr.: .... og deres inngivelsesdag ....

Hvis utskilt søknad: Den opprinnelige søknads nr.: .... begjært inngivelsesdag ....

Deponert kultur av mikroorganisme:

☐ Søknaden omfatter kultur av mikroorganisme. Oppgi også deponeringssted og nr. ....

Utlevering av prøve av kulturen:

☐ Prøve av den deponerte kultur av mikroorganisme skal bare utleveres til en særlig sakkyndig, jfr. patentlovens § 22 åttende ledd og patentforskriftenes § 38 første ledd

Angivelse av tegnings-figur som ønskes publisert sammen med sammendraget

Fig. nr. ....

PS-101-6000-12.2001

PATENTSTYRET

02-07-11*20023336

Seamless IP mobility across security boundaries

Abstract

A method, an arrangement and a computer program product, for providing seamless IP mobility across a security boundary between two domains is described. The invention comprises a novel architecture of known network infrastructure components along with enabling client software on the user device. The specific client software as well as the novel architecture represents the invention. Unlike state-of-art today, the method is based on the combined use of independent IP mobility systems in each of the two domains. The key to the invention is client software being able to operate with both mobility systems simultaneously. Moreover, communication takes place in such a way that the ordinary remote access security solution is in control of all access to the secure home domain of the user. The resulting method provides secure and seamless IP mobility in any domain with independent choice of mobility and security technologies. The invention does not require any significant changes (adaptations nor extensions) to any IP mobility or security technology beyond existing or upcoming standards. Nor does it require any significant changes to existing infrastructure components. The mobility client software is the only component that is affected, thus making the method client-centric, as opposed to a network-centric approach that is otherwise the alternative. The invention applies both for the current IPv4 family of standards as well as the forthcoming IPv6 family of standards. The invention applies in particular for the Mobile IP and IPSec VPN standards but is not restricted to these technologies. The invention is applicable for any IP mobility and IP security protocols as long as they are based on the same set of underlying principles.

Inventors: Klovning, Espen, (Oslo, Norway); Bryhni, Haakon, (Oslo, Norway); Nilsen, Frode Beckmann (Oslo, Norway);
Assignee: Birdstep Technology ASA, Oslo, Norway
Application no:
Filed:

Curennt U.S. Class:
Intern'l Class:
Field of Search:

Claims
What is claimed is:

A method, an arrangement and a computer program product, providing seamless IP mobility across a security boundary between domains.

Background and terminology
The large family of IP protocols constitutes the foundation for the development of the Internet. Today the Internet is based on version 4 of the protocol family (IPv4). In the future it expected to gradually be replaced by version 6 of the protocol family (IPv6).

IP mobility is an enhancement that has gain interest in recent years. Different IP mobility protocol proposals exist both for IPv4 and IPv6. Making the Internet mobile has obvious advantages compared to the legacy mobile networks that are tailored for voice communication only. Seamless IP mobility refers to the case when the user application is transparent to network changes. This is in contrast to ordinary IP when the application session breaks if the user changes his point of attachment to the network. The predominant IP mobility technology today is Mobile IP [6, 13] that exists both for version 4 and version 6.

The key parts of a Mobile IP system are Mobile IP client software on the user terminal and a Mobile IP Home Agent (HA) in the network infrastructure. The terminal with the client software is commonly referred to as a Mobile Node (MN). The home agent controls the topological correct address of the mobile node (called home address) and maintains a binding list with the current location of a mobile node (called care-of-address). The mobile node signals to the home agent its current care-of-address. This happens either directly, or optionally via an intermediate Foreign Agent (FA) if one exists in the local environment. The home agent sets up a forward tunnel to redirect traffic from the topologically correct home address to the current care-of-address. The tunnel arises from packet encapsulation performed by the home agent. For reference, any non-mobile host is referred to as a Correspondent Node (CN).

Seamless IP mobility finds its most important application together with remote access IP security solutions like IPSec VPN [7]. A remote access VPN solution consists of a VPN client on the user terminal and a VPN gateway (VPNGW) in the infrastructure. Together the VPN client and gateway employ both tunneling and encryption to maintain communication from a secure domain to a terminal that is remotely connected from an insecure location in a different domain. The VPN solution is usually the only way to reach components inside the secure domain.

State-of-art today is to leverage on the version 4 protocol family used on the Internet and to combine Mobile IP with IPSec VPN, thus facilitating the first generation architecture for seamless and secure IP mobility. The most prominent application is the enterprise environment in which the intranet represents the secure domain and the Internet represents the insecure domain. The potential of this combination is great since it gives nomadic workers less hassle and increased productivity as they are on the road. The most challenging part of the combined architecture is to maintain seamless operation also across the security boundary between the enterprise and the outside world. Any application started by a user while on the intranet should survive as the users moves outside. At the same time, the VPN solution must be employed while outside. Inside the secure domain the VPN solution should be turned off.

This document describes a novel architecture together with corresponding client software that can be used to solve the problem of seamless IP mobility across security boundaries. The proposed method is more general and has better characteristics than currently known alternatives.

In this document the term **architecture** is used to denote a combined structure that comprises both a Mobile IP part and a VPN part. The Mobile IP part in itself is referred to as a **system**. The VPN part is in contrast refer to as a **solution**. This is for clarity only. These terms could have been used interchangeably.

A VPN solution and a Mobile IP system both require client software on the mobile node. The term **client software** is generally used if the specific meaning is clear from the context. Otherwise the qualifiers "mobility" and "security" are added to distinguish the client software parts.

State-of-art architectures
As suggested by figure (1) and figure (2) there are basically two opposite ways of deploying a Mobile IP system together with a VPN solution to realize a combined architecture. Both architectures are well known to the standardization bodies [1] as well as in the industry in general [3,4,5]. The starting point is in either case a secure domain (105) separated from an insecure domain (107) by a security gateway (104). The gateway defines the boundary between the domains and the inside corresponds to the secure side. For simplicity the reader may think of an enterprise network (secure inside domain) connected to the Internet (insecure outside domain). The secure domain comprises one or more IP networks under the same administrative control. The secure domain is also referred to as the home domain of associated users. The insecure domain comprises IP networks that are controlled by different administrative entities. The security gateway (aka: firewall) is responsible for packet filtering as well as remote access to the secure domain from outside. Gray shading is used throughout this document to signify a secure domain with its constituent components.

In figure (1) the Mobile IP home agent (102) is located outside the secure domain, eg. in the de-militarized zone (106) of an enterprise environment. The Mobile IP system does in this case operate outside to the secure domain and its role is to mobilize the VPN solution. The system offers a fixed address for the VPN tunnel end-point, thus the VPN tunnel survives any network change made by the mobile node (103). The encrypted VPN packets are carried as payload in the Mobile IP system and the VPN solution is referred to as overlaid. This is illustrated in figure (1) by the VPN tunnel (123) being embraced by the Mobile IP tunnel (124). The native traffic (125) destined for the corresponding node (101) on the inside becomes apparent in the secure domain after decapsulation from the VPN tunnel. The Mobile IP components (102) and (103) are in this case drawn in a hatched gray/white pattern to signify that these components are associated with the secure domain (gray) but are otherwise operating in an insecure environment (white).

In figure (2) the home agent (102) is located inside the secure domain (105). The Mobile IP system does in this case operate on the inside and can only be reached from outside over the VPN solution, thus the home agent is drawn in a solid gray color. The VPN tunnel (123) does now embrace the Mobile IP tunnel (124), corresponding to the opposite sequence of figure (1).

The architecture shown in figure (2) suffers from the fact that the remote VPN tunnel end-point at the mobile node (103) will change at every network change. Accordingly, the VPN tunnel (123) needs to be re-established at every handover. This will in turn require re-establishment also of the Mobile IP tunnel. On the other hand, there are some disadvantages also with the architecture shown in figure (1). First, since the home agent (102) is deployed on the outside the home address of the mobile does not belong to the secure domain (105).

Consequently, the user must use his VPN solution from all locations, also when he is inside his otherwise secure home domain. For the same reason, when the mobile node is inside the secure domain traffic to any other corresponding node (101) on the inside will loop via the outside home agent (102). Both effects are unfortunate since they impose a performance penalty. Finally, there is a potential security risk since the home agent (102) needs to set up dynamic forwarding tunnels through the firewall (104) to arbitrary mobile nodes inside the secure domain. Normally, a tunnel through a firewall is allowed only if both end-points are static and if the traffic through the tunnel is subject to encryption both ways.

To overcome the disadvantages just mentioned it is possible to make hybrid architectures that combine the characteristics from figure (1) and (2). Figure (3) and figure (4) illustrates two different approaches. The architecture shown in figure (3) is at the core of some vendor-specific offerings available in the industry [4,5]. At the time of this writing the architecture shown in figure (4) is proposed for standardization to facilitate multi-vendor implementations [2]. Realizations of the architecture in figure (4) are currently not available from any vendor but are expected to become available in the near future.

In figure (3) the home agent (102) and the VPN gateway (104) are implemented in a single-box. The home agent is in this case drawn partly in solid gray and partly in a hatched pattern to indicate that it belongs to the secure domain yet at the same time it is accessible from outside without using the VPN solution. The operating model is essentially the same as in figure (2) but the architecture in figure (3) does not suffer from the same disadvantages. On the other hand, the single-box solution (102) gives restricted flexibility and scalability since the home agent component and the VPN components are always co-located. The fact that both components must be provided by the same vendor may also be a disadvantage. Many enterprises have already made existing investments in either Mobile IP or VPN components. An integrated single-box solution prevents these companies from leveraging their previous investments.

The architecture shown in figure (4) is based on the same principles as in figure (3) but it is implemented as a multi-box solution. The single-box solution is here replaced by three components; the inside home agent (102), the VPN gateway (104) and a proxy agent (108). The proxy agent is located in a de-militarized zone (106) outside the secure domain and it is basically the equivalent of a home agent and a mobile node implemented in the same box. Its role is to be an intermediate relay for signaling and packet forwarding between the real mobile node (103) and the inside home agent (102). The rationale of the proxy agent is to handle signaling and packet forwarding in a secure way by working in harmony with the VPN gateway and the internal home agent. The proxy agent must be located in a DMZ since embracing firewall policies must protect it. The proxy agent (108) is drawn partly in solid gray and partly in a hatched pattern to indicate that the inside home agent (102) trust it yet at the same time the proxy agent is accessible from outside without using the VPN solution.

Even if the architecture shown in figure (4) is an improvement over the one shown in figure (3), giving increased flexibility and also accommodating a multi-vendor environment, the architecture of figure (4) also has some unfortunate characteristics.

First, the proxy agent (108) requires specific protocol considerations as well as network design considerations. This is much due to the fact that both the VPN gateway (104) and the proxy agent (108) are involved in routing the same set of mobile node addresses, with the risk of conflicts or design flaws. These boxes must also be located on the same subnet, thus

reducing deployment flexibility. The fundamental idea is to ensure that both signaling and data packets between the mobile node (103) and the actual home agent (102) will always traverse the proxy agent (108). In the end, the proxy agent is a vulnerable component since the security of the architecture in figure (4) depends totally on a correct setup. On one hand, the proxy agent, the VPN gateway and the home agent must all work in harmony. On the other hand, the proxy agent must be protected by the firewall. In this complex environment any mis-configuration represent a security risk.

Secondly, the proxy agent is a novel component that must be developed by the industry before the architecture shown in figure (4) is ready for deployment. Moreover, extra capabilities are required both from the VPN gateway, the home agent and the mobile node that are either beyond the base protocols standard or beyond what is currently the state-of-art in the industry. Accommodating these changes will take time too. In sum, the architecture of figure (4) demands an effort from the whole industry before it is ready for deployment.

Finally, according to Amdahl's law [11] the performance of the proxy agent (108) and the VPN gateway (104) is linked. Increasing the performance of the total system requires as carefully balanced upgrade of the performance of its constituent's components. The inner home agent (102) and the proxy agent (108) are also linked since the proxy is designed to be the primary handling agent. A mobile node on the inside will attempt a registration with the proxy agent first, and only subsequently register with the inner home agent (102) if the proxy tells the mobile node to do so. This makes the proxy agent a critical component reducing the overall reliability of the total system. If the proxy agent becomes unavailable the whole system breaks down.

State-of-art client software
The current art of client software, including both the Mobile IP part and the VPN part, has impact on how a combined architecture can be realized. Taking a Mobile IP client vendor like Birdstep as the starting point, figure (5) and figure (6) show the prevailing implementation principles. These principles apply for today's state-of-art terminals like laptops and PDAs, running a state-of-art operating system like Microsoft Windows. These terminals and operating systems make a distinction between user space (119) and kernel space (120). User applications (121) run in user space. This is in contrast to key network components like a TCP/IP stack (118), drivers for LAN/WLAN adapters (110) or drivers for PPP dial-connection (111) that come as a native part of the operating system in kernel space. Note finally that the principles embodied in figure (5) and figure (6) assumes the overlaid VPN model from figure (2), (3) and (4). Hence, the VPN driver (117, horizontal) level is above the Mobile IP driver level (115, horizontal) in kernel space. The extensions of these software components into user space (115 and 117 vertical) represent the corresponding daemon parts.

The difference between figure (5) and (6) is how the VPN software is implemented. As suggested by figure (5), some vendors add a virtual VPN adapter (112) to the client software to host locally the address from the secure domain that is associated with the VPN connection. A virtual VPN adapter is just like any other adapter to the overlaying driver levels. However, since there is no physical connection on a virtual VPN adapter the VPN software (117) must take care of routing traffic to one of the physical adapters (110, 111). As suggested by figure (6) other VPN client implementations does not have a virtual VPN adapter. These solutions will rather maintain in the VPN gateway the association between a particular VPN connection and an address from the secure domain. The gateway will in this case also effectively perform a Network Address Translation (NAT) operation.

In either case, the Mobile IP client software (115) uses its own virtual mobility adapter (112) to maintain the invariable Mobile IP address. The user space part of the Mobile IP software (115, vertical part) is a daemon that determines what physical adapter represents the currently optimal connection. The role of the Mobile IP driver part (115, horizontal part) is to shift packet traffic accordingly between the virtual mobility adapter and the currently active physical adapter. The dashed vertical lines in figure (5) and figure (6) suggest how the different adapter instances are visible at all levels all the way up to the user application. The solid vertical line parts that connect the different levels in kernel space represent the packet shifting. Hence, these lines show how packets actually flow. Traffic to/from the user application are associated with the VPN adapter if one exists. Otherwise, the traffic is associated with the Mobility adapter. In the former case the VPN daemon will perform the required packet shifting between the VPN adapter and the Mobility adapter.

Client software for the architecture shown in figure (2), which is the opposite of the overlaid VPN architecture from figure (1), is realized along similar lines as those shown in figure (5) and figure (6). First, the drivers (115) and (117) must swap position in the stack. Second, packet shifting is less of an issue since the Mobile IP client will then always send its traffic over what is defined to be the default route by the VPN solution.

Please note that the hybrid architecture shown in figure (4) effectively assumes that the Mobile IP address and the inside VPN address associated with a mobile node (103) is the same address. For this reason, the hybrid architecture will only work if the VPN client does not have its own virtual adapter. A VPN client using a virtual VPN adapter will lead to an address conflict with the virtual mobility adapter used by Mobile IP client.

There are two classes among the state-of-art VPN solutions today regarding the interface seen by the user. Some vendors support a background-monitoring model in which the VPN client is resident and always ready for action. Any attempt to send traffic to what is defined to be the secure domain will automatically lead to encryption. Request for user authentication pops up automatically whenever needed. Other vendors depend on an explicit dial-type model in which the user himself must establish the VPN tunnel when communication with the secure domain is required. Some vendors support both operating models. Note also that most vendors support both full-tunnelling and split-tunnelling configurations in their VPN solutions. In the former case all traffic will be sent via the secure domain even if it destined for an outside location. With split-tunnelling traffic to outside locations can be sent directly from the mobile node when it is itself outside.

Note finally that a Mobile IP client can work with overlaid VPN solutions both in split-tunnelling and full-tunnelling mode. In the latter case the only traffic that is allowed to bypass the security solution is the Mobile IP signalling protocols.

Rationale of invention
The architecture in figure (4) represents state-of-art for seamless IP mobility across a security boundary today. The fact that the Mobile IP system works **across** the security boundary in **parallel** with the VPN gateway is unfortunate, however. The proxy agent represents a vulnerable component that must be carefully designed and deployed in harmony with the VPN gateway, the inside home agent and the embracing firewall.

The invention described in this document has its origin as a better approach than the architecture shown in figure (4) to the same problem. The proposed method combines IP mobility and IP security technologies in novel way to solve the problem more generally. The key to the invention is to use independent IP mobility systems at each side of the security boundary rather than one system spanning the boundary. The two mobility systems are in turn cleanly separated by the intermediate IP security solution. The role of the inner mobility system (running inside the security domain of the user) is to make user applications transparent to network changes internally. The role of the outer mobility system is to make the remote access security solution transparent to network changes externally. Together the two mobility systems and the remote access security solution facilitate seamless IP mobility also across the security boundary.

As previously pointed out the architecture from figure (4) suffers from the fact that the VPN solution and the Mobile IP system are closely interlinked in various aspects. In the end, this is all due to the fact that both parts operate in the same address domain and that they are involved in routing of the same set of mobile node addresses. The novelty of the proposed method is to make a split at this point by using a different set of addresses for mobility handling on each side of the security boundary. Since each of the three constituent parts operates in separate address domains their components can be deployed at any location within each domain. This is in contrast to the architecture shown in figure (4).

The invention represents a best-of-breed approach including the equivalent of both architectures shown in figure (1) and figure (2) at the same time. The gain of making a clean separation of mobility and security is a method that is more versatile and applicable in a wider scope. The proposed method is client-centric rather than network-centric. A client-centric approach requires no changes in the existing infrastructure except the introduction of a new mobility system. Since this second system is standards based anyway, the client-centric approach is easier and faster to deploy than a more involved network-centric approach. The method has less side-effects and less stringent deployment requirements than the alternatives known today. The scalability is also better since there are no implicit dependencies. Each system is deployed individually and can scale (and perform) independently as required. The reliability of the combined architecture is also better than the architecture from figure (4) since there are no dependencies between the parts. In particular, the inner system is fully operational even if the outer system and the security gateway components are temporarily unavailable. Moreover, the inner system will still be accessible from outside even if the outer system is temporarily down.

The invention provides secure and seamless IP mobility in any domain with independent choice of IP mobility and security technologies. The method does not require any changes (adaptations nor extensions) to any IP mobility or security technology beyond existing or upcoming standards. Nor does it require any changes to existing network design and infrastructure components. The mobility client software is the only component that is affected, hence being the enabling component. The method applies both for the current IPv4 family of standards as well as the forthcoming IPv6 family of standards. The method applies in particular for the Mobile IP and IPSec VPN standards but is not restricted to these technologies. The method is applicable for any IP mobility and IP security protocols as long as they are based on the same few underlying principles.

Description of the method

The general method is best described when casted in a more specific setting. Hence, this section describes the invention assuming that Mobile IP is used for mobility handling and that an IPSec VPN solution is used for remote access to the secure domain. The generality and extendibility of the method is described in a subsequent section.

As suggested by figure (7), the method is based on the combined use of independent Mobile IP systems in the secure domain (105) and insecure domain (107), respectively. The gray shaded home agent (102, gray) inside the secure domain represents the inner system. The hatched home agent (102, hatched) outside the secure domain represents the outer system. The enabling component of the method is the Mobile IP client software running on the user device that can operate with **both** Mobile IP systems simultaneously. The dual capability is indicated by the partly solid gray and partly in a hatched pattern on the mobile node (103).

It is assumed that the user's ordinary VPN solution is in control of **all** access to the secure home domain. In particular, operation of the inner Mobile IP system takes place over the VPN solution; ie. the mobile IP tunnel (124, thin) for the inner system is embraced by the VPN tunnel (123). This corresponds to the architecture shown in figure (2). Operation of the outer Mobile IP system is in contrast according to the architecture shown in figure (1); ie. the VPN tunnel (123) is embraced by the Mobile IP tunnel (124, thick) of the outer system. Hence, the proposed method is a best-of-breed approach including both legacy architectures at the same time. This is reflected by the fact that there are three levels of tunnels involved when the mobile node connects from the outside. The native user traffic (125) is in the end carried by the Mobile IP tunnel (124) of the inner system.

The role of the inner Mobile IP system is to make user applications transparent to network and address changes inside the secure domain. Note that this includes the transition case when the mobile node moves from the inside to the outside. Then a VPN tunnel will be established that is basically a prolonged arm of the secure domain. A key feature of the method is to use the internal address that is associated with the VPN tunnell (123) as the care-of-address in the binding list of the inner home agent (102, gray). This inner care-of-address will **not** change as long as the mobile node (103) connects from the outside via the VPN tunnel. At this point the outer Mobile IP system becomes important. The role of the outside system is to make the VPN tunnel itself transparent to network and address changes on the outside. In sum, the two Mobile IP systems take different roles but works naturally together isolated by the intermediate VPN solution.

The inner system (102, gray) is always in use for a mobile node. The outer system (102, hatched) is in use only when the mobile node is outside. In this case the operation of the inner system becomes trivial for the reason just explained. Moreover, the inner system will always operate in co-located mode when the mobile node (103) is on the outside. Otherwise, both Mobile IP systems can operate both in co-located mode and in foreign agent mode. Foreign agents that are deployed on the inside will be used by the inner system. Foreign agents that are deployed on the outside will be used by the outer system.

The issue of reverse tunneling for the two systems is discussed in a subsequent section on addressing details.

There are no specific requirements on the address range that is used for mobile nodes in the inner system. Any address range can also be use for the outside system. The use of mobile

node addresses in the outer system is of a more dynamic nature, though, since this system is only in use as long as the mobile node connects from the outside. In this period the terminal is actually equipped with two mobile node addresses, one from each system. Whereas the mobile node address from the inner system is permanent during operation, the mobile node address from the outer system can be allocated dynamically as needed. The only requirement is that these two addresses must never be the same. If private addresses are used for the outside system reverse tunneling must be used as always.

Any subnet inside the secure domain can be designated as the home network for the inner Mobile IP system. A designated home network for the outer system is more of an open issue. If the outer home agent is deployed in the DMZ (106) of an enterprise, the system will usually run with a virtual home network. It is very uncommon to allow hosts to connect directly to the DMZ. An interesting opportunity arises if the enterprise has already a WLAN on its premises connected to the firewall on a separate segment. Such a configuration is common today since a WLAN is insecure and should not be connected directly inside the secure domain of the enterprise. Rather, the VPN solution is used to connect over the WLAN from outside. As suggested by the figure (8) the WLAN (109) will in this case be the perfect place for the outer home agent (102, hatched). The important observation is that any mobile node (103) connecting to this network will effectively be transparent to the outer Mobile IP system since it is the designated outer home. As suggested by figure (8) a mobile node will in this case connect to the inner system over the VPN solution only. Saving the overhead of the embracing outer tunnel (124) contributes to better performance, which is particularly important in this case since it is reasonable to expect that many users will spend a large portion of their working day in this wireless environment.

To summarize this section, consider the following list of the key characteristics of the two mobile IP systems under various conditions

Inner Mobile IP system:
- Is always in use
- Physical home network on internal subnet
- User's application binds to the home address of this system, hence application transparency and seamless operation, both internally and across the security boundary

Mobile node is **inside:**
- The VPN solution is turned off.
- Operation in the standard way (according to Mobile IP) using the inner home agent

Mobile node outside
- The VPN solution is turned on
- Signalling messages are exchanged with the inner home agent over the VPN tunnel.
- care-of-address in binding list of the inner home agent is always the address of the Mobile node's address from the secure domain (maintained by the VPN solution).
- Operation of inner system becomes trivial

Outer Mobile IP system
- Is used as required
- Physical home on externally connected enterprise WLAN

Mobile node is **inside**
- The outer system is not in use

Mobile node **outside**

- Outer system is required
- Binding list of outer home agent holds the address of the mobile node's current external address
- The VPN tunnel binds to the home address of the outer system, hence VPN transparency

Dual Client implementation

The enabling component of the invention is Mobile IP client software that can operate with two independent Mobile IP systems simultaneously, and in such a way that these systems are separated by an ordinary VPN solution. In the following this kind of client will be referred to as a **dual mobility client.** This is in contrast to a singular mobility client that is otherwise state-of-art today. When executed in a computer device of a mobile node, the software computer product of the invention establishes the arrangement of the invention. When executed in a computer device of a mobile node, the software computer product of the invention effects the execution of the method of the invention.

Implementation of a dual client on a device running a state-of-art operating system (like members of the Microsoft Windows family) is based on the same implementation principles as previously outlined in figure (5) and figure (6). Two enhancements are then made; first, another virtual mobility adapter is added to host the invariable address of the second Mobile IP systems. Next, a new Mobile IP driver level is included above the VPN driver level. The resulting kernel-space architecture is shown in figure (9), assuming that the VPN client uses its own virtual VPN adapter (114). The new upper Mobile IP driver (116) together with the new virtual mobility adapter, now called the intranet adapter (113) accounts for the inner Mobile IP system. The lower Mobile IP driver (115) together with the original virtual mobility adapter (112) accounts for the outer system. The intermediate VPN driver level (117) isolates the two systems and takes care of all security in the ordinary way. The arrows shown in figure (9) suggest how traffic is shifted on its way through the driver stack. This particular figure corresponds to the case when the mobile node is outside the secure domain and both systems are in use.

The seamless operation results from the fact that the application level above the upper Mobile IP driver (116) relates to the intranet adapter (113) that will always be present and available as a transport end-point.

Figure (10) shows the kernel-space architecture of a dual Mobile IP client when the intermediate VPN solution does not have its own virtual VPN adapter. Again it represents the case when the mobile node is outside the secure domain. The sequence of arrows signifying packet shifting is slightly different due to the missing virtual VPN adapter.

As already pointed out the inner system is always in use whereas the outer system is in use only when the mobile node is outside the secure domain. Moreover, the operation of the inner system is trivial when the mobile node is on the outside since the inner care-of-address is then always the intranet address associated with the VPN connection and will not change. When the mobile node is inside the secure domain the operation of the inner system becomes more complex as it must handle true network handovers and address changes. This complexity is the same as for the outer system when it is in use. Figure (11) shows the resulting kernel-space architecture that applies when the mobile node is on the inside. The components (116, 117, 112) become transparent or idle in this case as explained below.

The key point regarding implementation is that the upper Mobile IP driver (116) can be disabled when the mobile node is on the inside. At the same time the lower Mobile IP driver starts supporting the inner system rather than the outer system. Hence, the implementation of the dual client rests on the fact that there is a **context switch** in the operating environment of the lower Mobile IP driver (115). A part of this context shift is to let the lower driver start operating on the intranet adapter (113) rather than the mobility adapter (112). As a consequence, true mobility handling with packet shifting to physical adapters is only implemented in the lower Mobility driver. This driver is used for the inner or the outer system, depending on if the mobile node is on the inside or on the outside. In the latter case, the upper Mobile IP driver starts it operation to maintain the inner system until the mobile node returns to the secure domain. For reasons already explained, the complexity of this driver is trivial compared to the lower driver.

The VPN solution (117) is also turned off when the mobile node is on the inside. Any virtual VPN adapter used by the VPN solution will then disappear from the operating environment. Hence, the kernel-space architecture will be the same regardless of the kind of VPN solution being used.

The preceding description is based on the assumption that the dual client is implemented on a "open platform" device having a modular operating system environment comprising both user space and kernel space. Moreover, the concept of a driver and an adapter is central to the discussion. Other devices, and in particular embedded devices, may have operating environments that are very different. If there is no modular architecture the dual client must be implemented as a monolithic system. Depending on the capabilities of the host device such a system can be implemented in either user space or kernel space. In some cases it might even be included as an integral part of the host operating system and the default run-time environment.

The flow-chart in figure (12) represents a more abstract view of the traffic-flow principles that have just been outlined in a specific setting. The general principles must underpin any implementation of a dual client, regardless of the specific characteristics of the particular host device. The flow-chart assumes that neither ordinary reverse tunnelling nor NAT traversal is needed when the Mobile IP client connects directly to the inner system. The role of reverse tunnelling is otherwise discussed in a subsequent section on addressing.

Domain detection
The principles outlined in the previous section comprise the traffic flow part of a dual client implementation. The control part (normally implemented in a daemon) must be enhanced accordingly to also include support for a second mobile IP system. Moreover, this capability must be made optional so that it can be activated as required (for reasons already explained). The final piece is to let the daemon initiate the context-switch, depending on if the mobile node is inside or outside the secure domain. This is based on a **domain detection** mechanism that is carefully designed. Since this decision in the end governs the on/off status of the VPN client, it is paramount that the detection algorithm is reliable and cannot be compromised. At the same time it is important that the decision can be made quickly to reduce the performance penalty when the mobile node moves between networks, and in particular when it moves across the security domain.

The domain detection mechanism is concerned about the follow two transitions:

- When the mobile node was already connected from outside and a new inside connection is recognized (or suspected).
- When the mobile node was already connected at the inside and a new outside connection is recognized (or suspected).

In either case, the new connection can be over the same adapter as the previous connection or it can be over a different adapter.

In the end, the only reliable evidence that the mobile node is connected to either of the two domains via one of its physical adapters is when a registration with the appropriate home agent succeeds over that adapter. The reliability rests on the security associations between the home agent(s) and the mobile node that is already a part of the Mobile IP system(s). The brute-force approach is to always attempt registrations with both agents whenever domain detection is required. The downside of this approach is poor handover performance, in particular across the security boundary. On one hand this is because registrations may not get through to the appropriate home agent when issued from the opposite side of the security boundary. Handling pending registrations contribute to the complexity of the operation in the daemon and should be avoided whenever possible. On the other side, if a registration succeeds domain detection is complete but the active adapter has also then changed. The latter is really an undesired side-effect if the adapter turns out not to be the optimal. The recent registration will then be overridden immediately resulting in a performance penalty.

The need here is to have a reliable home agent service discovery mechanism that is not linked to a real registration. A spin-off activity of this invention is to try to standardize such a mechanism. It the meantime the performance-degrading effects can be overcome by first making conjectures about location that are based on less reliable but faster domain detection mechanisms with no side-effects. Operation can then proceed more quickly with a fairly high probability that the conjecture in the end turns out to be correct. If the initial conjecture was wrong, the brute-force approach will in the end lead to the correct conclusion. The less reliable domain detection mechanisms that are covered by the invention are as follows:

- Compare observed IP address information with pre-configured address ranges used inside the secure domain
- Utilize information in control messages like ICMP (including agent advertisements) that is a constituent part of the normal traffic flow on the network.
- Perform an unreliable home agent detection by sending a registration with deliberately wrong security association. This will effectively "smoke out" any home agent if it is reachable without actually performing a registration.
- Utilize network context history that is recorded and stored during operation. Such information is collected for foreign agents, default gateways, WLAN access points, etc. using their layer two identifiers (like MAC address) as the basis for indexing.

Addressing details
Three different parts of the proposed architecture in figure (7), ie. the inner Mobile IP system, the intermediate VPN solution and the outer Mobile IP system, result in three levels of addresses and tunnel encapsulations. This can be hard to understand until it is explicitly spelled out. Equipped with an overall description of the method together with the principles for a dual client implementation, this section describes the details regarding addressing.

Figure (13) shows the various components involved together with the address designators that are used for the different components. Note that the mobile node (103) is equipped with four different addresses, corresponding to the two mobile IP systems (102, gray and 102, hatched),

the VPN solution (104) and finally the physical address of the network to which the mobile node is currently connected (107 in this case). The V-address on the mobile node suggests that there is a local VPN adapter in this case. If there is no local VPN adapter the V-address will not exist locally. The T-addresses for the outer home agent and the VPN gateway refers to interfaces that are reachable from the outside.

The use of reverse tunnelling for each of the two systems is closely related to addressing. Reverse tunneling might be required for the inner system depending on the characteristics of the VPN solution. VPN solutions using a virtual VPN adapter may enforce strict filtering policies, either on the client side, or on the gateway side that effectively will prevent any traffic over the VPN connection with a source address that is different from the V-address allocated to the remote connection. Employing reverse tunneling will ensure that the topologically correct source address is used for reverse traffic. This is nothing different from using reverse tunneling to overcome ingress filtering in the outside domain.

For VPN solutions not using a virtual VPN adapter there is a similar rationale for using reverse tunneling but this time only to overcome any actions taken on the client side. The VPN gateway should not expect any particular source address since it has not distributed any to the local device. Many of these gateways will instead perform a NAT operation to convert between a topologically valid address on the inside (the equivalent of a V-address) and the actual address used for the connection on the outside (the T-address will be used). In this case the care-of-address registered in the inner home agent (ie. the T-address) may become topologically invalid if it is not routable inside the secure domain. To overcome this problem NAT traversal of Mobile IP [10] must be supported by the inner system. A specific form of tunneling is then used for the inner system, both in the forward and the reverse direction. This is nothing different from using the same mechanism in the outer system to overcome NAT devices in the outside domain.

Except from the reasons just explained reverse tunneling should in general not be required for the inner system when connected on the inside. The assumption is that S-addresses are accommodated as valid source addresses on any inside network and that there are no NAT devices on the inside. Mobile nodes on the inside can send traffic directly in the reverse direction. The need for reverse tunneling in the outer system is "as usual". It is required either if private mobile node addresses are used or if ingress filtering is being performed on any local access.

Consider now the sequence of steps taking place in an externally attached mobile node for a packet that is destined for a corresponding node inside the enterprise. This is shown in figure (14) also suggesting what parts taking place at the different levels. For simplicity, it is assumed that the correct registrations have already been completed both for the inner and the outer Mobile IP systems. Assuming that reverse tunnelling is required for the outer system, the packet that eventually leaves the mobile node (over the active adapter) is destined for the outer home agent. The structure of this packet shows clearly the outer Mobile IP reverse tunnel encapsulation, then the VPN tunnel encapsulation, and finally the inner Mobile IP reverse tunnel encapsulation. The reason for the inner reverse tunnel is to ensure a correct topological source address corresponding to the VPN address.

Consider next the sequence of steps that takes place in the outer home agent, the VPN gateway and eventually the inner home agent as the same packet traverses these components on its way to the corresponding node inside the secure domain. This is shown in figure (15).

If a VPN solution without a VPN adapter is used, there are minor changes to the picture. The V-address will then not exist in the mobile node and the T address is temporarily used in its place. Normally, when the packet arrives at the VPN gateway the correct V-address will replace the T-address, effectively performing a NAT operation. For this reason a UDP based reverse tunnelling might need to be inserted. This is discussed in more detailed later.

Packets originating from a corresponding node and destined for an outside mobile node follows the opposite path of what has just been described. The resulting figure is not shown explicitly but follows directly from considering figure (14) and figure (15) in revers and at the same time swap the addresses in all source/destination pairs. The reverse tunnel encapsulations from figure (14) and (15) will then be replaced by corresponding forward tunnel encapsulation from the home agents, of course.

A similar exercise for the packet flow between a corresponding node and mobile node inside the secure domain is trivial since it reduces to ordinary Mobile IP operation in a single system with no VPN solution involved.

Scope of the invention
The description of the invention has up to this point been casted in a specific setting; ie. an enterprise deploying its own Mobile IP service in an IPv4 environment using an existing IPSec VPN gateway. The proposed method emerged as a novel solution to this problem but has in itself a much wider scope of applicability. First, the involved technologies and components can be replaced by other technologies and components as long as the replacements share a few common characteristics with the components and technologies in the original description. Secondly, the method can be applied in many other deployment scenarios to solve other problems than the original one. This section focuses on the first aspect. The next section is devoted to describing other deployment scenarios of interest.

In the end, the proposed method depends only on a few fundamental and general characteristics:
- That both mobility systems run over an IP transport infrastructure.
- That both mobility systems are based on a client-server model in which the server component (aka: agent) is the authoritative source for mobility handling.
- That the remote access security solution (if any) runs over an IP transport infrastructure.
- That the remote access solution is based on a client-server model in which the server component enforces the security policy at the domain boundary.
- That the remote access solution provides a secure IP transport facility between the client and the server.
- That the server component of the security solution is able to distinguish between individual remote connections. Further, that a unique address is associated with each such connection and that this address is routable from the inside domain.
- That the terminal device of interest has capabilities (hardware and software) facilitating implementation of a system with dual client capabilities.

These principles are already embedded in the base Mobile IP standards for both IPv4 and IPv6, hence it follows that the invention applies also when the base standards are accompanied by additional standards extending the base protocols. Note in particular that binding updates (that is a default part of Mobile IPv6 to support routing optimisation) is

naturally accommodated by the proposed method. The same applies for any (future) enhancements and improvements to the family of Mobile IP standards. As a particularly important example the upcoming standard on NAT traversal using UDP encapsulation [10] is supported. In fact, NAT traversal can be supported in each of the two systems individually. Likewise, the important upcoming Authentication Authorisation and Accounting (AAA) protocol extensions [12] of Mobile IP are supported in both systems.

The principles from the above list are also already embedded in the base IPSec standards, again including both IPv4 and IPv6. Consequently, the proposed method works with all options, extensions and additions included in both IPSec protocol standard families. Note in particular that the method works with IPSec both in tunnel mode and transport mode if each of those modes is otherwise applicable in the particular context.

The following (non-exhaustive) list gives some additional examples of what is covered by the proposed method.
- Other tunnelling mechanisms than those that are currently embodied in the Mobile IP standards. In particular, there is a gain for tunnelling header compression to reduce the overhead imposed by the three layer of encapsulations.
- Other IP mobility protocols than Mobile IP, if they comply with the principles from the above list.
- Other IP security protocols than IPSec, if they comply with the principles from the above list. In particular, the L2TP[8] and PPTP[9] protocols are both supported.
- The trivial "null" case with no security solution at all.
- Mixed systems using IPv4 and IPv6 in different parts (eg. IPv6 inside and IPv4 outside). The only assumption is that any of the known IPv4-IPv6 transition methods otherwise apply in the particular context. Moreover, the mobile node must be equipped with a dual-stack solution in this case.
- Other kind of client devices than laptops, PDAs and similar personal terminal devices. In particular, the method can be applied when the client side is implemented as a part of a mobile router that is capable of moving a complete network (with multiple users) rather than just a single user.

Deployments of the Method
The first application of the proposed method is to solve the known problem of seamless IP mobility across an enterprise security boundary in a new and more beneficial way. Even if the method emerged as a novel solution to this problem it has in itself a much wider scope of applicability. In this section two other deployment examples are briefly described. These are both solutions to problems that have not yet been addressed or solved by the industry. These problems are both conjectured to become important in the future.

Up to this point it has implicitly been assumed that both IP mobility systems are run and administrated by the same entity (the enterprise). The independence between the systems opens for a very interesting opportunity though, namely that the systems are operated by different administrative entities. An enterprise will itself always be responsible for the inner system since this system is part of the secure domain. The outer system is different and the enterprise can equally well exploit a mobility service from a public network operator. The public operator can on his side offer the same service to many enterprises. The business opportunity for the operator is to deploy home agents along the edges of the public network, close to where enterprises have their security gateways. An illustration of this idea is shown in figure (16). The operator can also offer dynamic allocation of outer home agent if his service

is linked to an AAA infrastructure. The enterprise user will then always use the most optimal agent in the area where his security gateway is located. This is a particularly important feature for a user from an advanced enterprise since he may then use different security gateways depending on his actual outside location. Distributed and redundant security gateway architecture is state-of-art today among large enterprises spanning multiple sites and countries. In sum, independent operation and administration of the two mobility systems facilitate new future business models among and across public operators and private enterprises. The only requirement is to include handling of two separate administrative interfaces in the dual mobility client.

Another interesting application of the proposed method is to build mobility solutions across networks with different version of the IP protocol. Today, Mobile IPv4 is the most mature technology. Mobile IPv6 is still being standardized but is conjecture to become increasingly important as the transition from IPv4 to IPv6 networks otherwise takes place. The situation on the Internet in the first phase of this transition period will be a number of IPv6 network islands immersed in a large sea if IPv4 networks. Building a mobility service across these network boundaries lends itself naturally to a dual client solution. The only assumption is that the terminal device comprising the mobile node has dual-stack architecture. Ie. a device including both a IPv4 and a IPv6 protocol environment and supporting both protocols at the same time. The current commercial implementations are usually dual stack implementations. The result is a transition method where the end-systems can reap the benefits of both Mobile IP v6 and Mobile Ipv4 in a mixed environment. The transition boundaries can also be security boundaries, of course, as long as the security gateway is also dual-stacked. For an enterprise two operational modes can be envisaged. The first operational mode where the inner IP mobility system is using Mobile IPv4, is applicable for enterprises and other organizations that would like to maintain their IPv4 based system while reaping the benefits of an Mobile IPv6 solution on the outside. The second operational where the inner IP mobility system is Mobile IPv6, is applicable for enterprises or organizations that have migrated to IPv6 but where the core network beyond the secured domain is still using the IPv4.

Summary of benefits and characteristics
This final section lists the key characteristics of the proposed method. Focus is on the contribution of the invention compared to what is otherwise state-of-art today. The list covers benefits of the method itself as well as how application of the method can be used to solve different problems in different circumstances.
- A best-of-breed approach supporting any of the well-know architectures represented by figure (1,2,4).
- Based on using two simultaneous but truly independent mobility systems isolated by an ordinary security solution
- The ordinary security solution is in control of all remote access to the secure domain, in particular the inner mobility system runs via the remote access solution if the mobile node connects from outside
- The two systems fulfil different roles; inner system is responsible for user application transparency. The outer system is responsible for transparency of the security solution itself.
- The design is open demonstrating independency between mobility and security
- The combined system offers total seamless operation on both sides as well as across the security boundary.
- Only the client side is affected. A dual mobility client is the enabling component.

- The complexity of the dual client software is not much higher than a singular client. First, the second mobility system is not always in use. Second, when it is in use the operation is almost trivial
- The client-centric approach (as opposed to a network-centric) facilitates easy, faster and cost-effective deployment.
- No specific network design requirements beyond the extra mobility system. Works with existing network designs and require no changes to existing infrastructure components.
- Facilitates multi-vendor deployment with independent choice of mobility system and security solutions. In particular, previous investments can be leveraged.
- The combined architecture is flexible and scales well since each of the three constituent components can be deployed individually and scale independently as needed.
- Requires no changes (adaptations nor extensions) to any mobility or security technology beyond existing or upcoming standards.
- Based only on a few fundamental principles that are already embedded in all pertinent IPv4 and IPv6 standards. Hence, method will work with all options, extensions and enhancements of these protocols.
- Method works also with other mobility and security protocols as long as they share the same set of basic characteristics.
- The inner and outer mobility systems can be operated by different entities; hence they do not need to be part of the same administrative domain.
- The method solves the known problem of seamless Mobile IP across an enterprise security boundary in a new and better way
- The method can be used to solve similar problems in a much wider scope. This includes new business models for IP mobility across public operators and private enterprises.

References
[1] Problem Statement for Mobile IPv4 Traversal Across VPN Gateways
    http://www.ietf.org/internet-drafts/draft-ietf-mobileip-vpn-problem-statement-00.txt
[2] Mobile IPv4 Traversal Across IPsec-based VPN Gateways
    http://www.ietf.org/internet-drafts/draft-adrangi-mobileip-vpn-traversal-02
[3] Birdstep Technology ASA
    http://www.birdstep.com/
[4] ipUnplugged AB
    http://www.ipunplugged.sw/
[5] Ecutel Inc
    http://www.ecutel.com/
[6] IP Mobility Support for IPv4, revised (RFC 3220)
    http://www.ietf.org/rfc/rfc3220.txt
[7] IPSec standards
    http://www.ietf.org/rfc/rfc2411.txt?number=2411
[8] Secure Remote Access with L2TP
    http://www.ietf.org/rfc/rfc2888.txt?number=2888
[9] Point-to-Point Tunneling Protocol
    http://www.ietf.org/rfc/rfc2637.txt?number=2637
[10] Mobile IP NAT/NAPT Traversal using UDP Tunnelling
    http://www.ietf.org/internet-drafts/draft-ietf-mobileip-nat-traversal-04.txt
[11] Amdahls law
    http://www.cis.temple.edu/~shi/docs/amdahl/amdahl.html
[12] AAA Registration Keys for Mobile IP
    http://www.ietf.org/internet-drafts/draft-ietf-mobileip-aaa-key-09.txt
[13] Mobile IPv6

Numbering plan (reference numerals) for figure components

101 = Corresponding Node (ordinary host)
102 = Mobile IP Home Agent
103 = Mobile IP Node
104 = Security Gateway (VPN + Firewall)
105 = Secure Domain (eg. private enterprise network)
106 = De-militirized Zone (DMZ)
107 = Insecure domain (eg. public network)
108 = Proxy Agent
109 = Wireless LAN network
110 = Ethernet LAN/WLAN Network Interface Card
111 = PPP connection
112 = Birdstep Mobility Adapter (outer system)
113 = Birdstep Intranet Adapter (inner system)
114 = 3rd party VPN adapter
115 = Birdstep Mobile IP driver (lower)
116 = Birdstep Mobile IP driver (upper)
117 = 3rd party VPN driver
118 = Operating System native TCP/IP stack
119 = User Space
120 = Kernel Space
121 = User Space Application
123 = VPN tunnell encapsulation
124 = Mobile IP tunnell encapsulation
125 = Native IP packet traffic

Figure 1

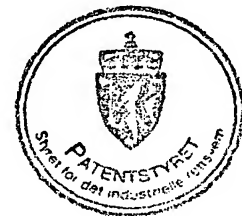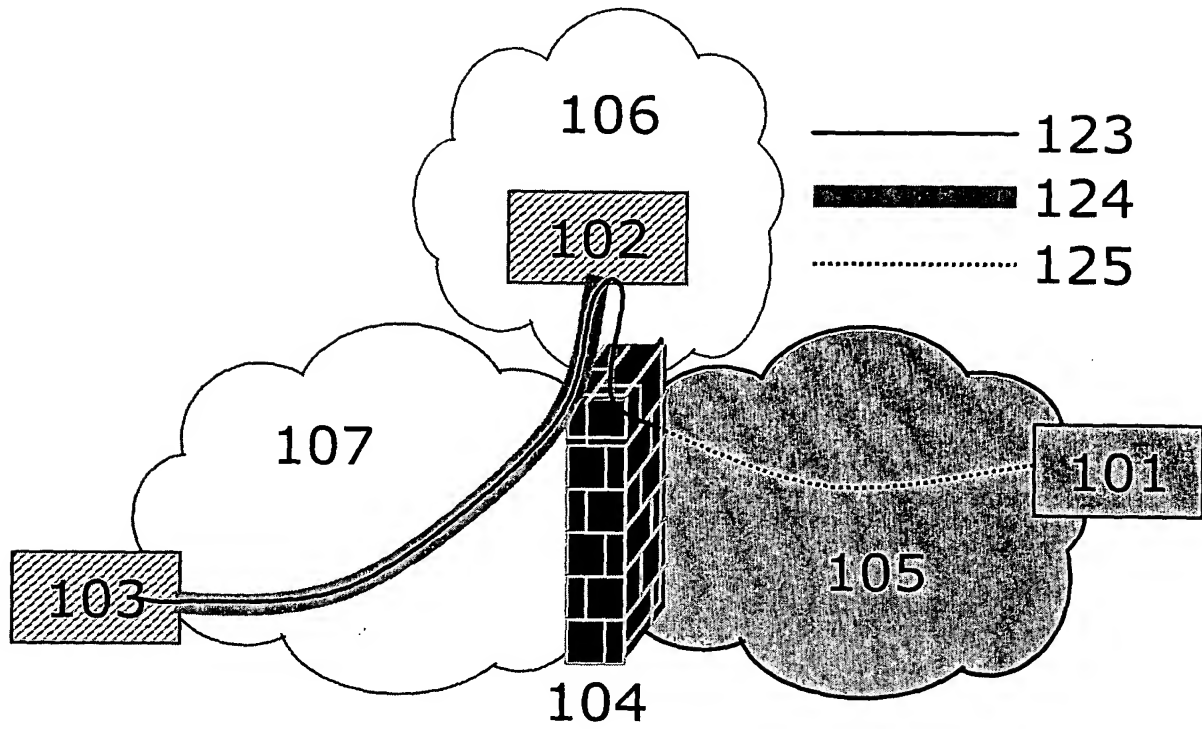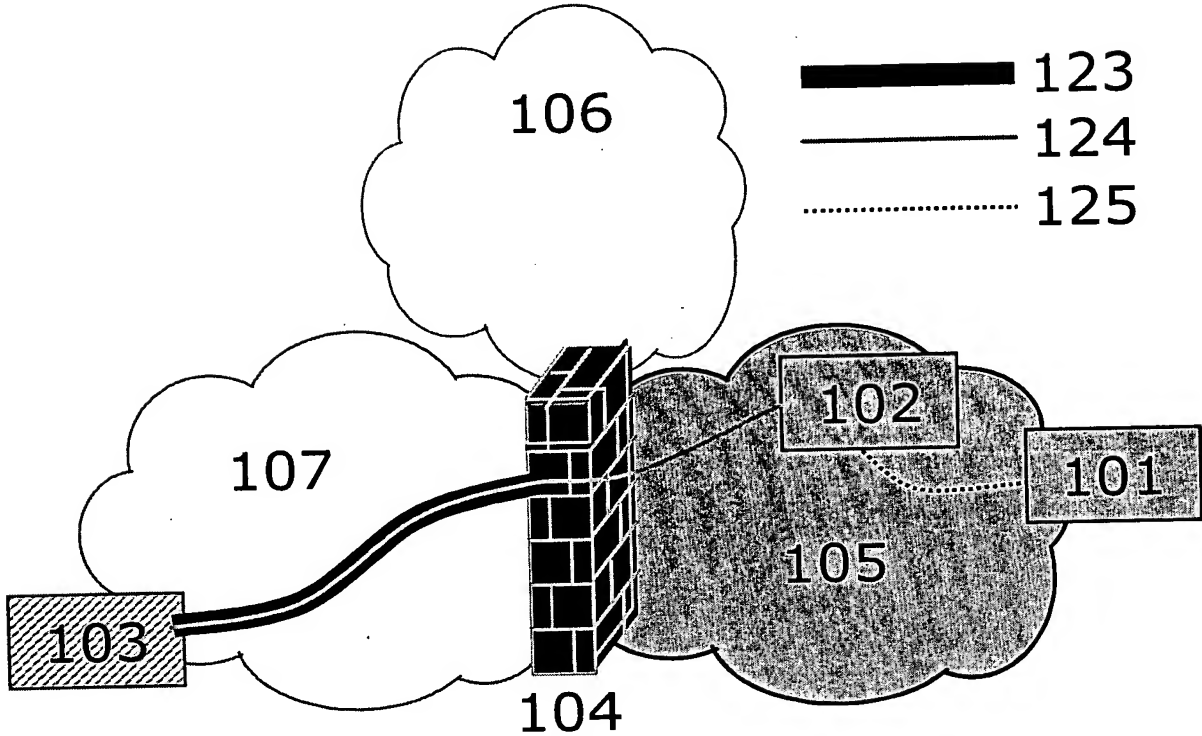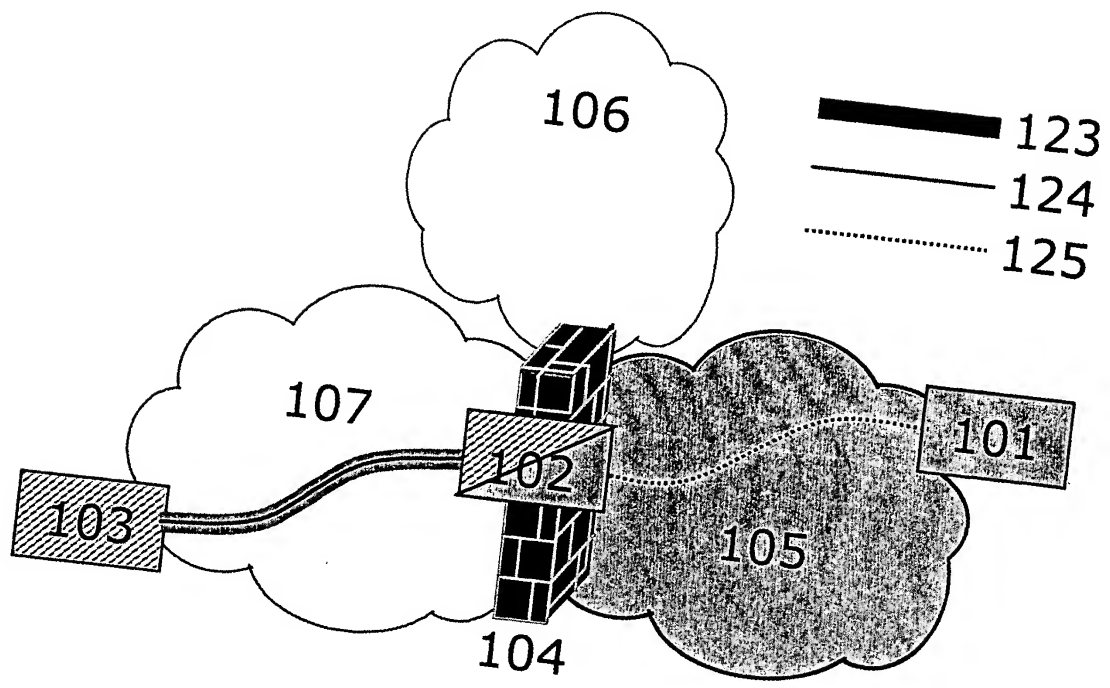**Figure 2**

Figure 3

Figure 4

Figure 5

119

120
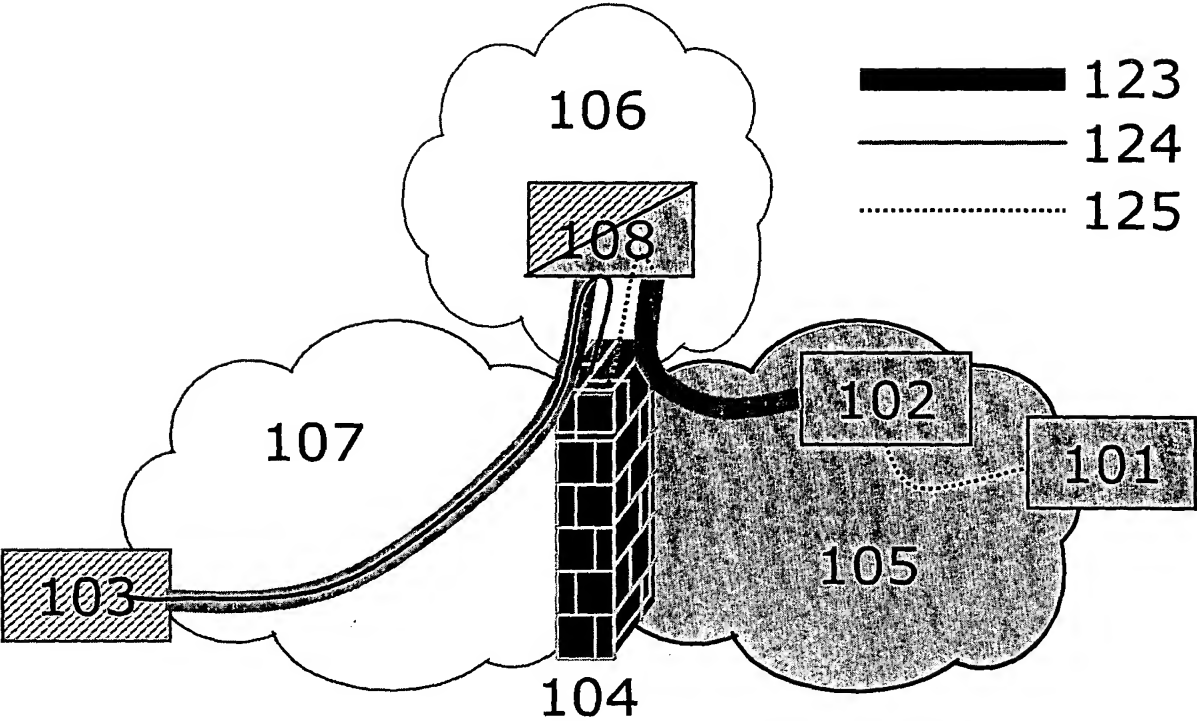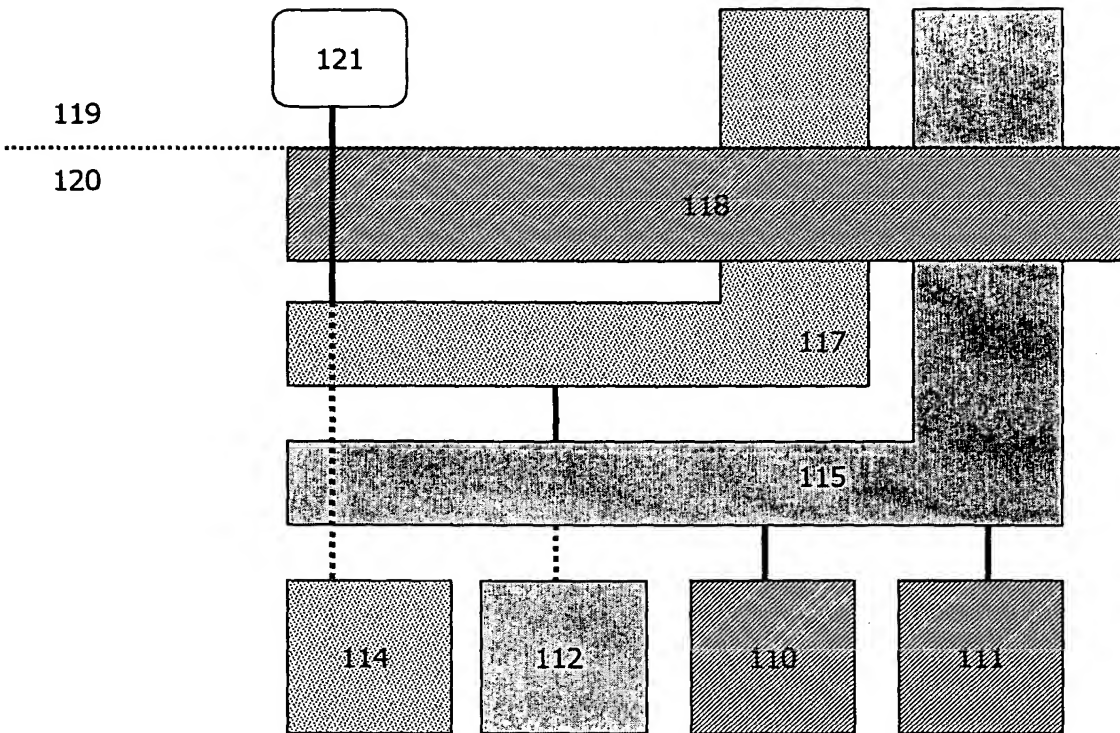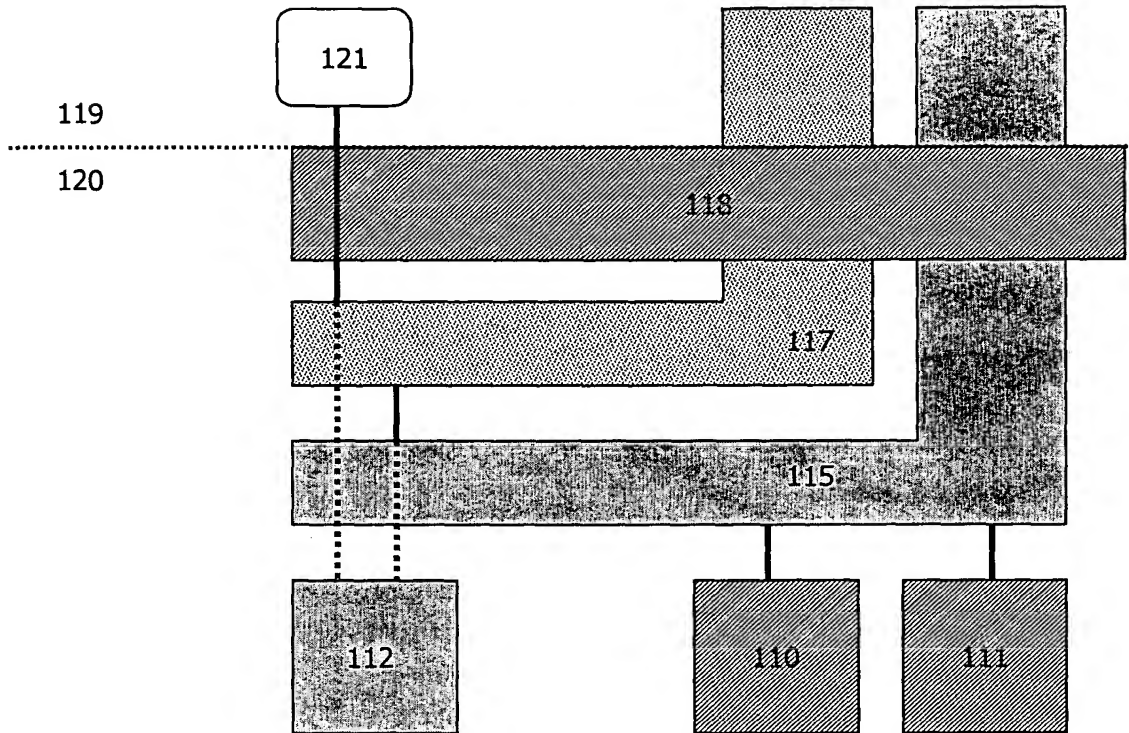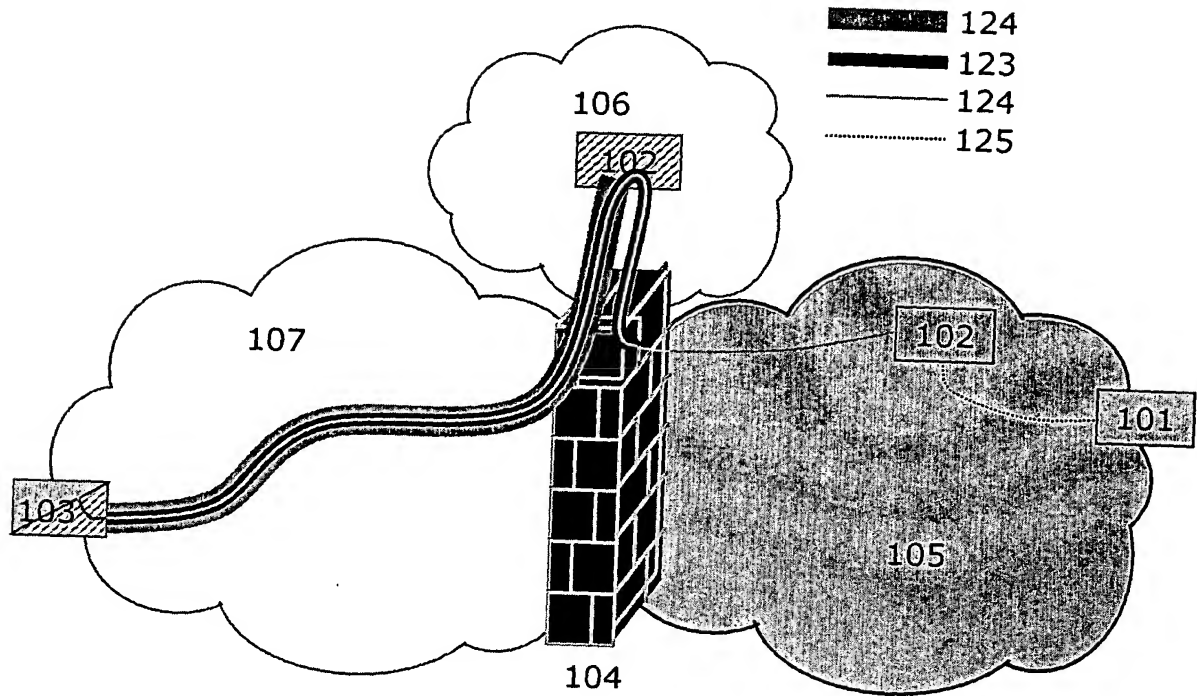
Figure 6

Figure 7

Figure 8.
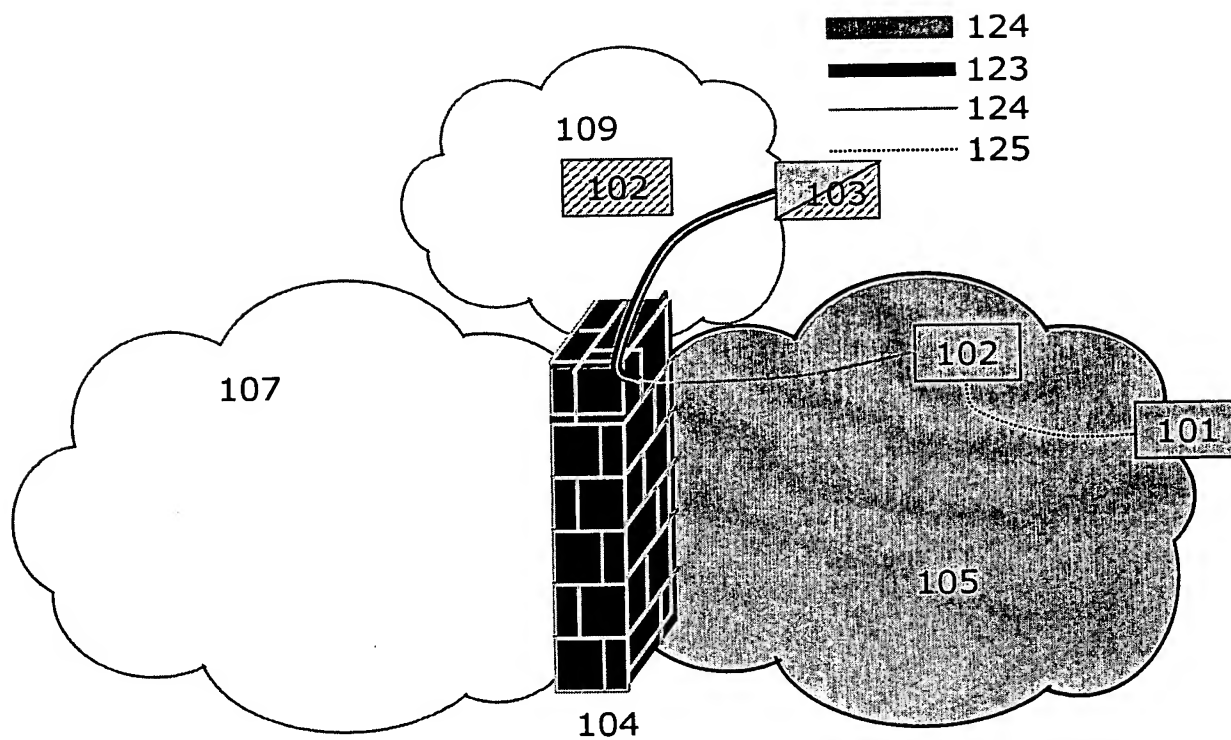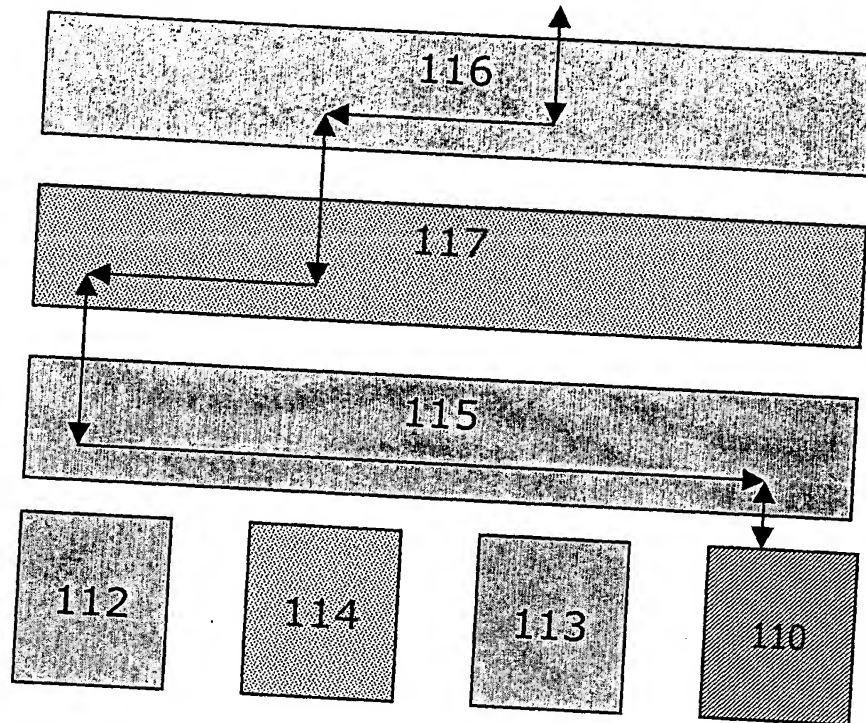
Figure 9

Figure 10

Figure 11

Figure 12

```
                        application
                          packet

                      bind source
                      address to
                    intranet adapter

                    outside secure
      No                 domain?              Yes

                                          local VPN
                                  No      adapter?      Yes

  Shift adapter:        shift adapter:            Shift adapter:
  intranet->active      intranet->mobility        intranet->VPN

                                                VPN ingress        Add reverse
                             NAT in             filtering?   Yes   tunelling using
                    No       VPNGW?                                VPN adapter as
                                                                   source
                              Yes
  VPN ingress        Add reverse       Add NAT friendly
  filtering?   Yes   tunelling using   reverse tunelling      VPN encrypt
                     mobility adapter   using mobility
                     as source         adapter as source
                                                             Shift adapter:
      No            VPN encrypt                               VPN->mobility

                                       Add VPN tunelling
                                       using mobility
                                       adapter as source

                                       Shift adapter:
                                       mobility->active

                                       Reverse tunnell
                              No        in outer          Yes
                                        system?

                                                          Add reverse
                        sendt                             tunelling using
                        packet                            active adapter as
                                                          source
```
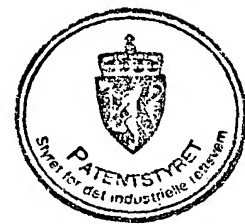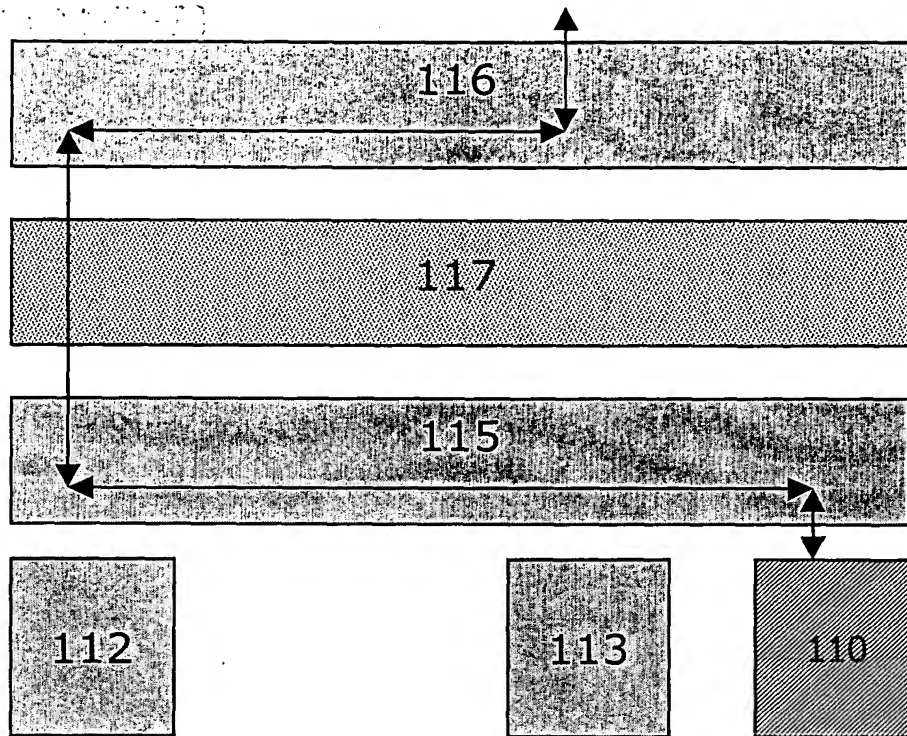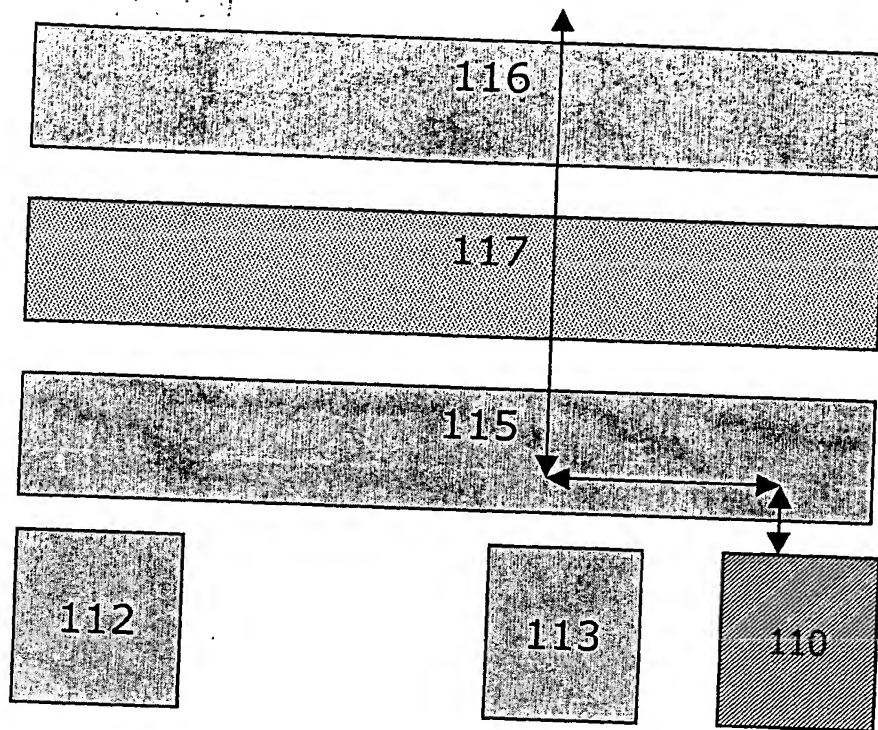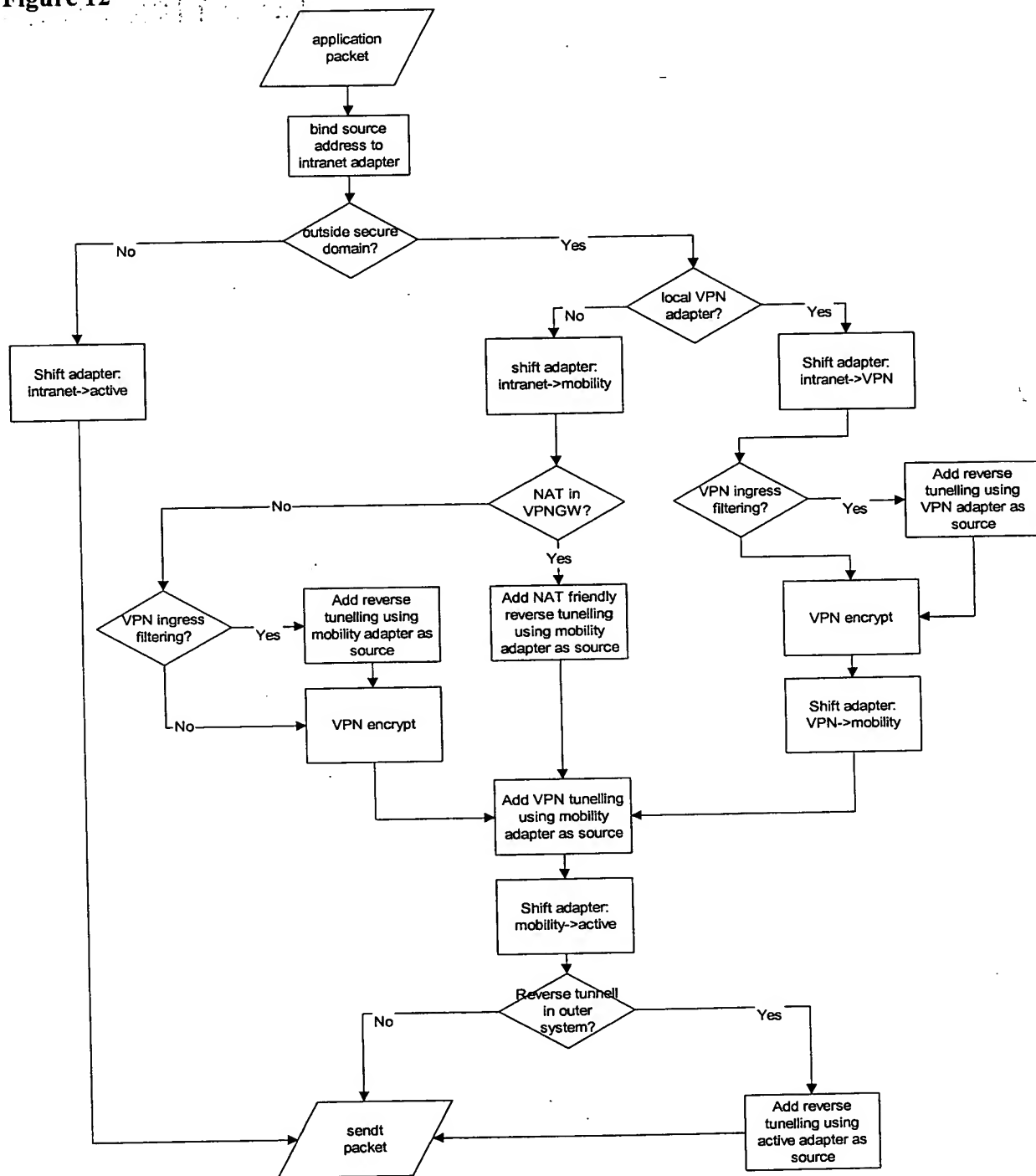
Figure 13

T$_1$: External MIP addr
V$_1$: VPN adapter addr
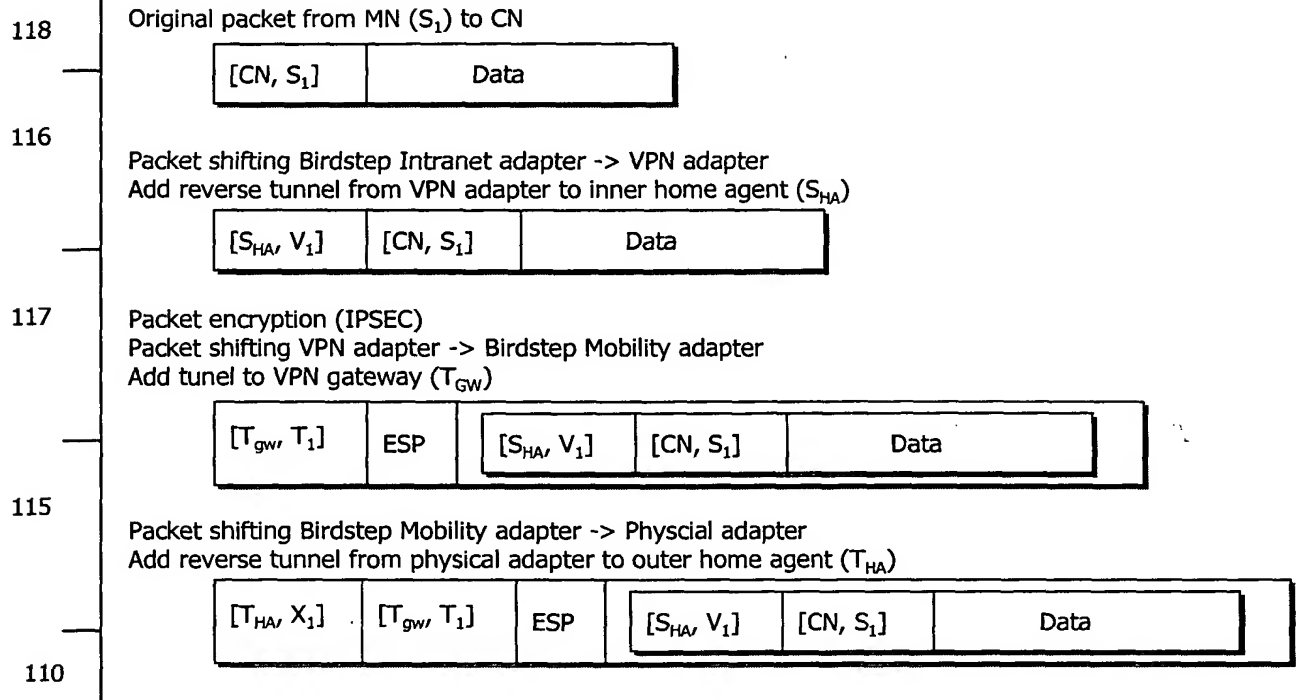X$_1$: Physcial addr
S$_1$: Intranet MIP addr

T$_x$: pool of external MIP addresses

103

T$_{HA}$ 102

T$_{gw}$

107

104

V$_x$: Pool of internal VPN addresses

102

S$_{HA}$

101

S$_x$: Pool of internal MIP addresses

105

Figure 14

118

Original packet from MN (S$_1$) to CN

[CN, S$_1$]	Data

116

Packet shifting Birdstep Intranet adapter -> VPN adapter
Add reverse tunnel from VPN adapter to inner home agent (S$_{HA}$)

[S$_{HA}$, V$_1$]	[CN, S$_1$]	Data

117

Packet encryption (IPSEC)
Packet shifting VPN adapter -> Birdstep Mobility adapter
Add tunel to VPN gateway (T$_{GW}$)

[T$_{gw}$, T$_1$]	ESP	[S$_{HA}$, V$_1$]	[CN, S$_1$]	Data

115

Packet shifting Birdstep Mobility adapter -> Physcial adapter
Add reverse tunnel from physical adapter to outer home agent (T$_{HA}$)

[T$_{HA}$, X$_1$]	[T$_{gw}$, T$_1$]	ESP	[S$_{HA}$, V$_1$]	[CN, S$_1$]	Data

110

Figure 15

Packet received by outer home agent (T_{HA})

$[T_{HA}, X_1]$	$[T_{gw}, T_1]$	ESP	$[S_{HA}, V_1]$	$[CN, S_1]$	Data
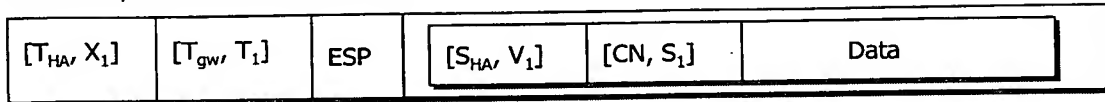
Decapsulate reverse tunnel
Packet forwarding to VPN gateway (T_{gw})

$[T_{gw}, T_1]$	ESP	$[S_{HA}, V_1]$	$[CN, S_1]$	Data
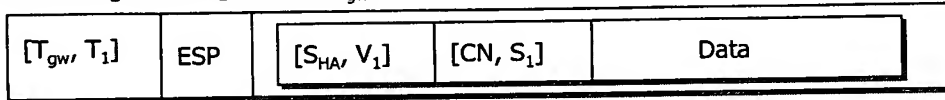
Decapsulate and decrypt in VPN gateway (T_{gw})
Packet forwarding to inner home agent (S_{HA})

$[S_{HA}, V_1]$	$[CN, S_1]$	Data

Decapsulate reverse tunnel in inner home agent (S_{HA})
Packet forwarding to corresponding node (CN)

$[CN, S_1]$	Data

Figure 16: